Challenges of Open Source Licensing in the spin-off of scientific software

Dr. Carsten Emde
Open Source Automation Development Lab (OSADL) eG

Why this topic?

- Electronic devices usually contain software.

Why this topic?

- Electronic devices usually contain software.
- New software is developed for newly developed electronic devices. This software is a combination of in-house developments and adaptations of existing software. The latter is usually **Open Source software**.

Why this topic?

- Electronic devices usually contain software.
- New software is developed for newly developed electronic devices. This software is a combination of in-house developments and adaptations of existing software. The latter is usually **Open Source software**.
- If a larger group of users is interested in the newly developed electronic devices, **spin-off companies** are sometimes founded to market the devices.

Which challenges?

- Open Source software grants you full rights to **immediately use** the software **on your own computer** ☺️.

Which challenges?

- Open Source software grants you full rights to **immediately use** the software **on your own computer** ☺️.
- However, when Open Source software is **copied and distributed**, the same general rules of copyright law apply as for proprietary software – namely, that a **valid copyright license** must be obtained ☹️.

Which challenges?

- Open Source software grants you full rights to **immediately use** the software **on your own computer** ☺️.
- However, when Open Source software is **copied and distributed**, the same general rules of copyright law apply as for proprietary software – namely, that a **valid copyright license** must be obtained ☹️.
- Although it is usually easier than for proprietary software, Open Source software also requires a **certain amount of effort** to obtain a license to copy and distribute the software.

Which challenges?

- Open Source software grants you full rights to **immediately use** the software **on your own computer** ☺️.
- However, when Open Source software is **copied and distributed**, the same general rules of copyright law apply as for proprietary software – namely that **valid copyright licenses** must be obtained ☹️.
- Although it is usually easier than for proprietary software, Open Source software also requires a **certain amount of effort** to obtain a license to copy and distribute the software.

IF THIS IS NOT CONSIDERED WHEN THE SOFTWARE IS INCLUDED, IT MAY BE COMPLETELY IMPOSSIBLE TO OBTAIN A LICENSE WHEN MARKETING

What is Open Source software?

- **“Open Source”** is not an English phrase, but a **technical term**.
  - ➔ Cannot be translated
  - ➔ The original form “Open Source” must always be used – irrespective in which context, country or language.
- Open Source software must fulfill certain **requirements**, for example given in the Open Source Definition (OSD) released by the Open Source Initiative (OSI, https://opensource.org/OSD).

What is Open Source software? (2)

Requirements for Open Source software

- A software is Open Source if the holders of rights allow everyone
  - ✔ to **unrestrictedly and unconditionally use**, **analyze** and **modify** the software
  - ✔ to **copy and distribute** the software under very liberal **conditions**.
- Otherwise it may not be called “Open Source”.
- In particular: Being available as **source code** does **not** automatically make a software **“Open Source”**.

Proprietary vs. Open Source software

Action	Proprietary software		Open Source software	
	What is needed?	By whom/ why?	What is needed?	By whom/ why?
Run the software		**Prohibited** by authors		**Permitted** by unilateral declaration of will of the authors
Analyze the software		**Prohibited** by copyright law		
Modify the software				

Proprietary vs. Open Source software

Action	Proprietary software		Open Source software	
	What is needed?	By whom/ why?	What is needed?	By whom/ why?
Run the software	End User License Agreement (EULA)	**Permitted** by EULA	*Access to the software*	**Permitted** by unilateral declaration of will of the authors
Analyze the software		**Prohibited** by copyright law	*Access to the software*	
Modify the software			*Access to the software*	

Proprietary vs. Open Source software

Action	Proprietary software		Open Source software	
	What is needed?	By whom/ why?	What is needed?	By whom/ why?
Run the software	End User License Agreement (EULA)	**Permitted** by EULA	*Access to the software*	**Permitted** by unilateral declaration of will of the authors
Analyze the software		**Prohibited** by copyright law	*Access to the software*	
Modify the software			*Access to the software*	
Copy and distribute unmodified software		**Prohibited** by copyright law		**Prohibited** by copyright law
Copy and distribute modified software				

Proprietary vs. Open Source software

Action	Proprietary software		Open Source software	
	What is needed?	By whom/ why?	What is needed?	By whom/ why?
Run the software	End User License Agreement (EULA)	**Permitted** by EULA	*Access to the software*	**Permitted** by unilateral declaration of will of the authors
Analyze the software		**Prohibited** by copyright law	*Access to the software*	
Modify the software			*Access to the software*	
Copy and distribute unmodified software	**License**	**Permitted** by license	**License**	**Permitted** by license
Copy and distribute modified software		**Prohibited** by copyright law	**License**	

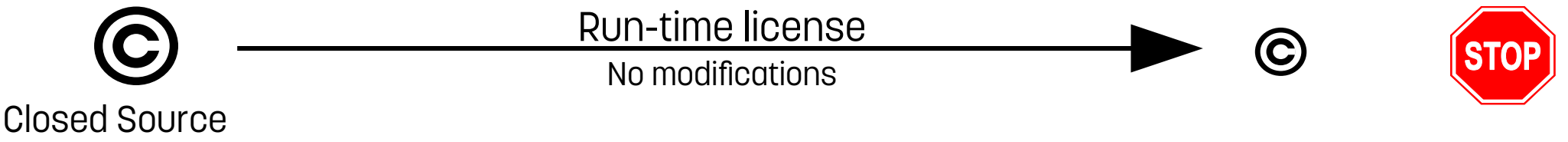
Proprietary vs. Open Source software

Action	Proprietary software		Open Source software	
	What is needed?	By whom/ why?	What is needed?	By whom/ why?
Run the software	End User License Agreement (EULA)	**Permitted** by EULA	*Access to the software*	**Permitted** by unilateral declaration of will of the authors
Analyze the software		**Prohibited** by copyright law		
Modify the software			*Access to the software*	
Copy and distribute unmodified software	**License**	**Permitted** by license	**License**	
Copy and distribute modified software		**Prohibited** by copyright law	**License**	

Tough control rules must be accepted

Liberal license obligations must be fulfilled

What is “Copyleft”?



What is “Copyleft”?

Run-time license

No modifications

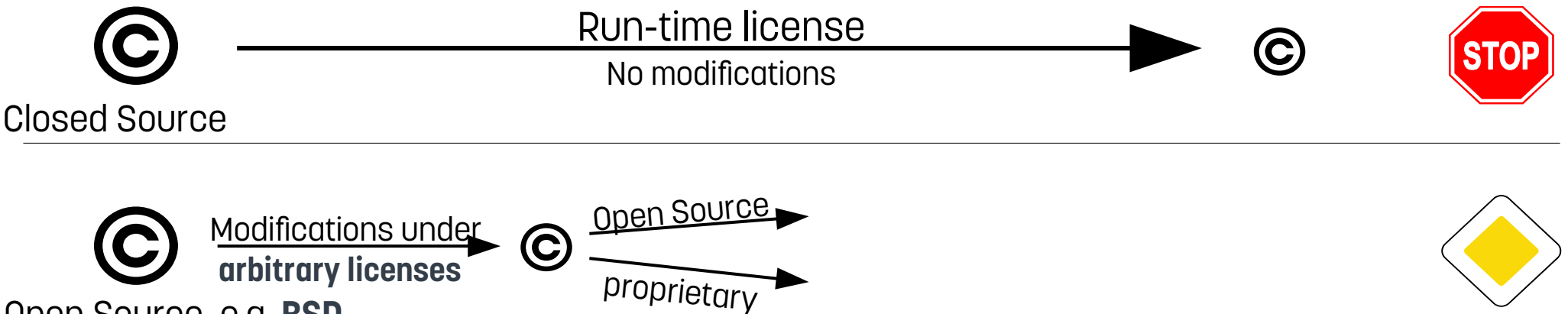
Closed Source

Modifications under

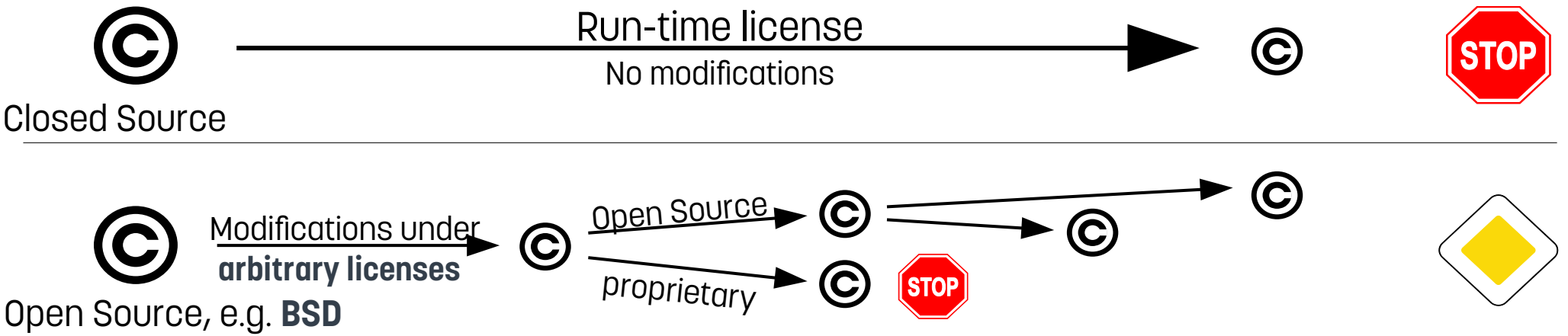
arbitrary licenses

Open Source, e.g. **BSD**

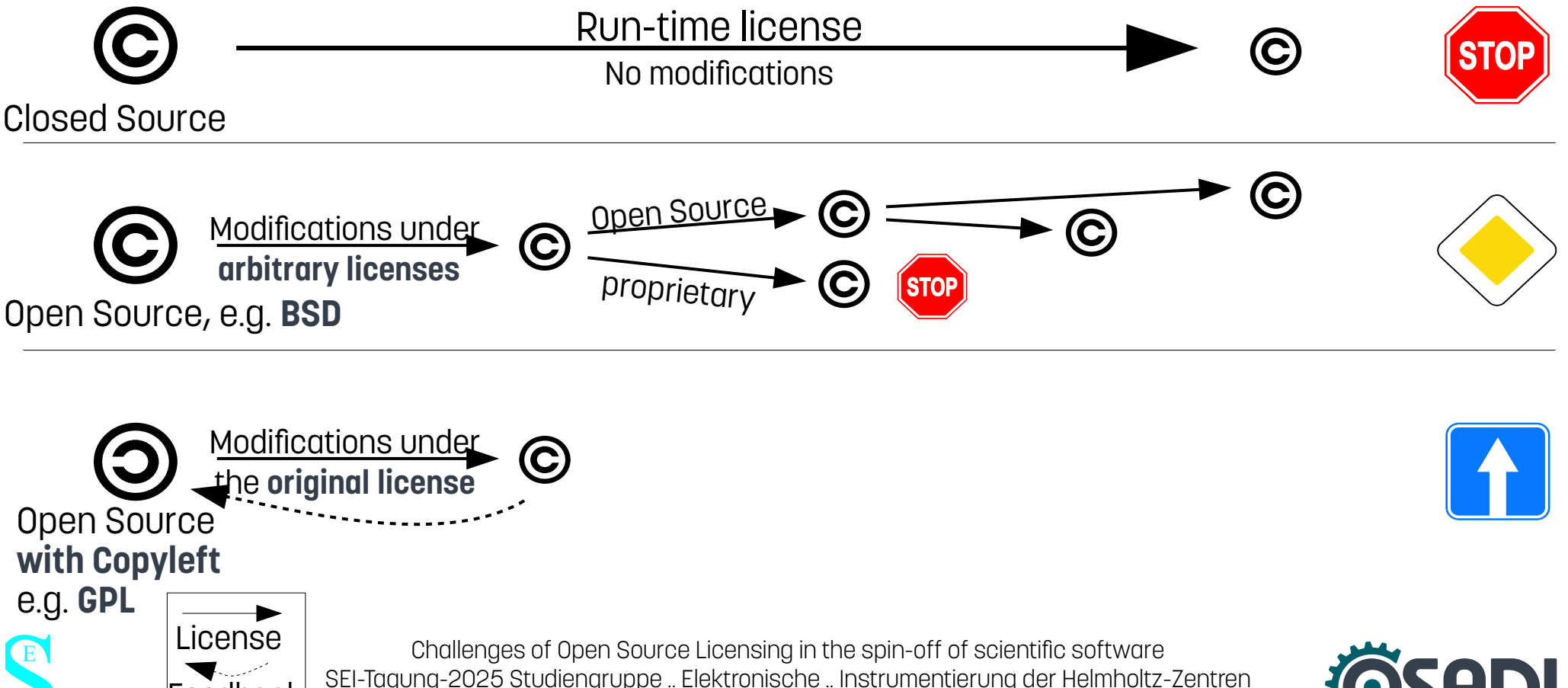
What is “Copyleft”?



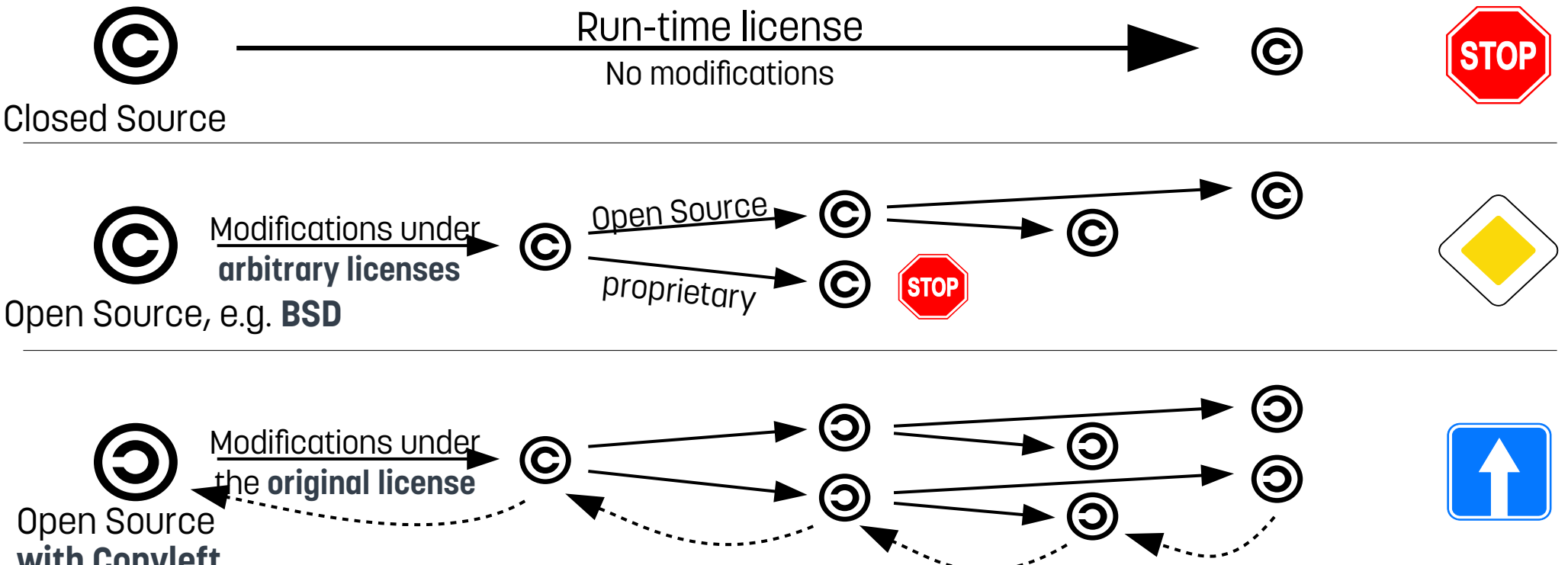
What is “Copyleft”?



What is “Copyleft”?



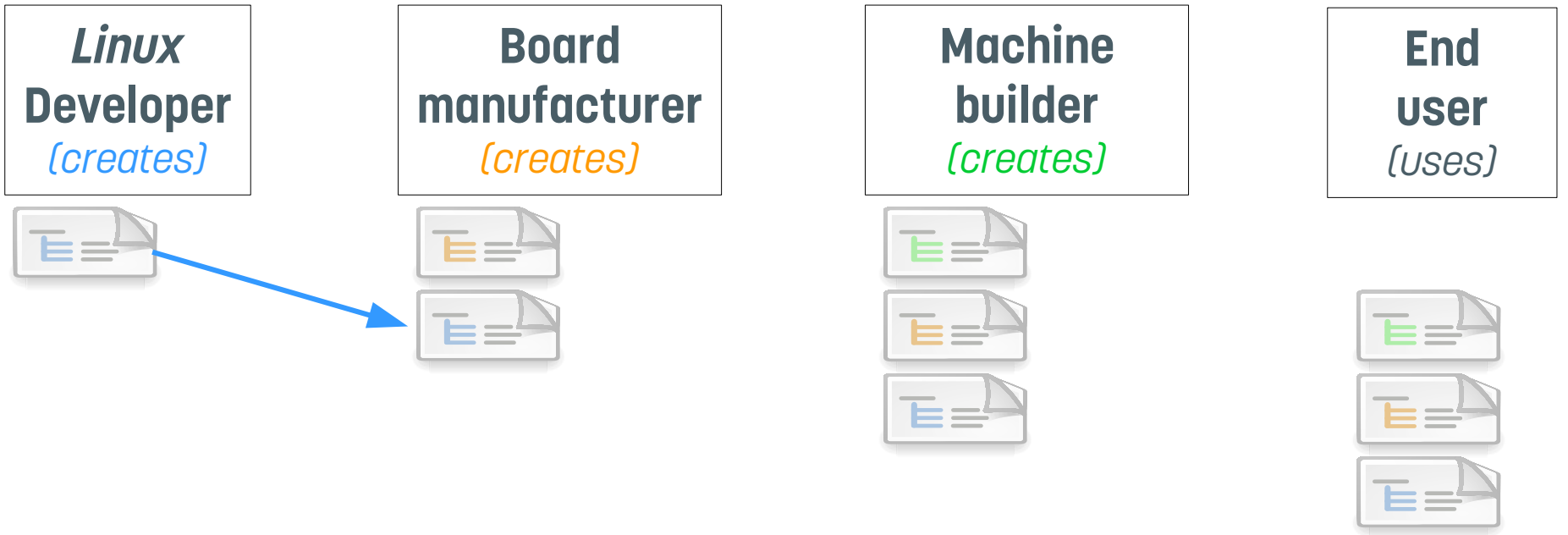
What is “Copyleft”?



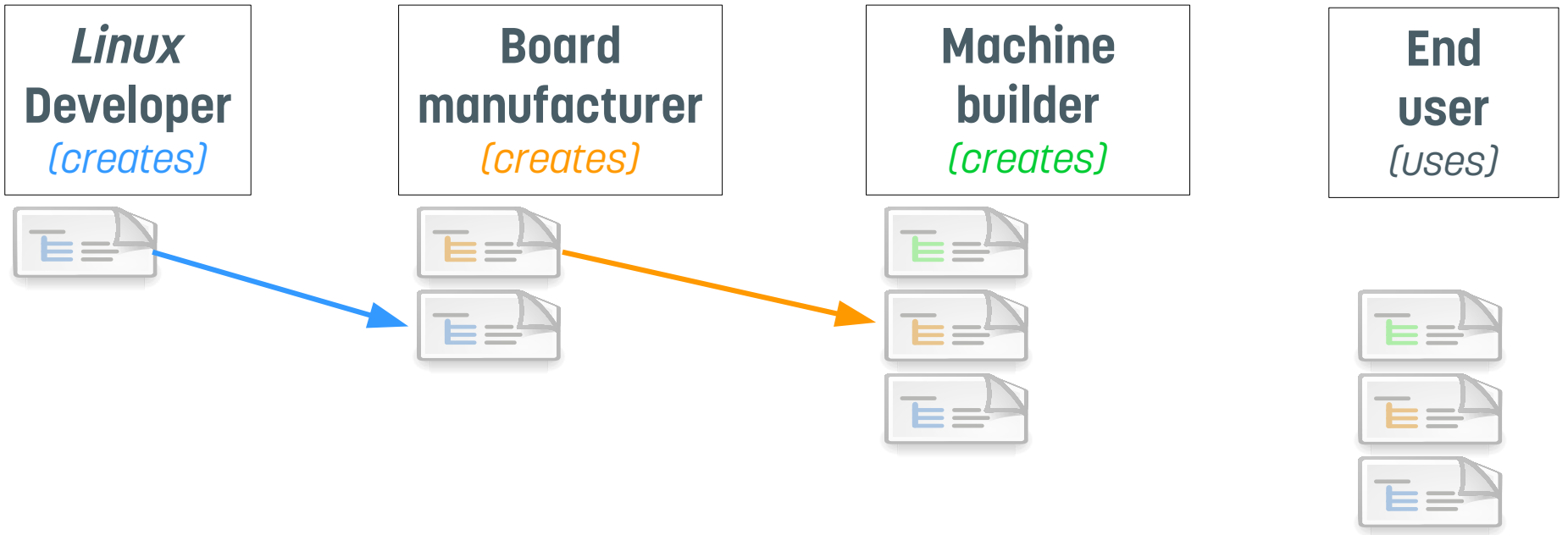
GPL software trade chain

Linux Developer *(creates)*	Board manufacturer *(creates)*	Machine builder *(creates)*	End user *(uses)*
			

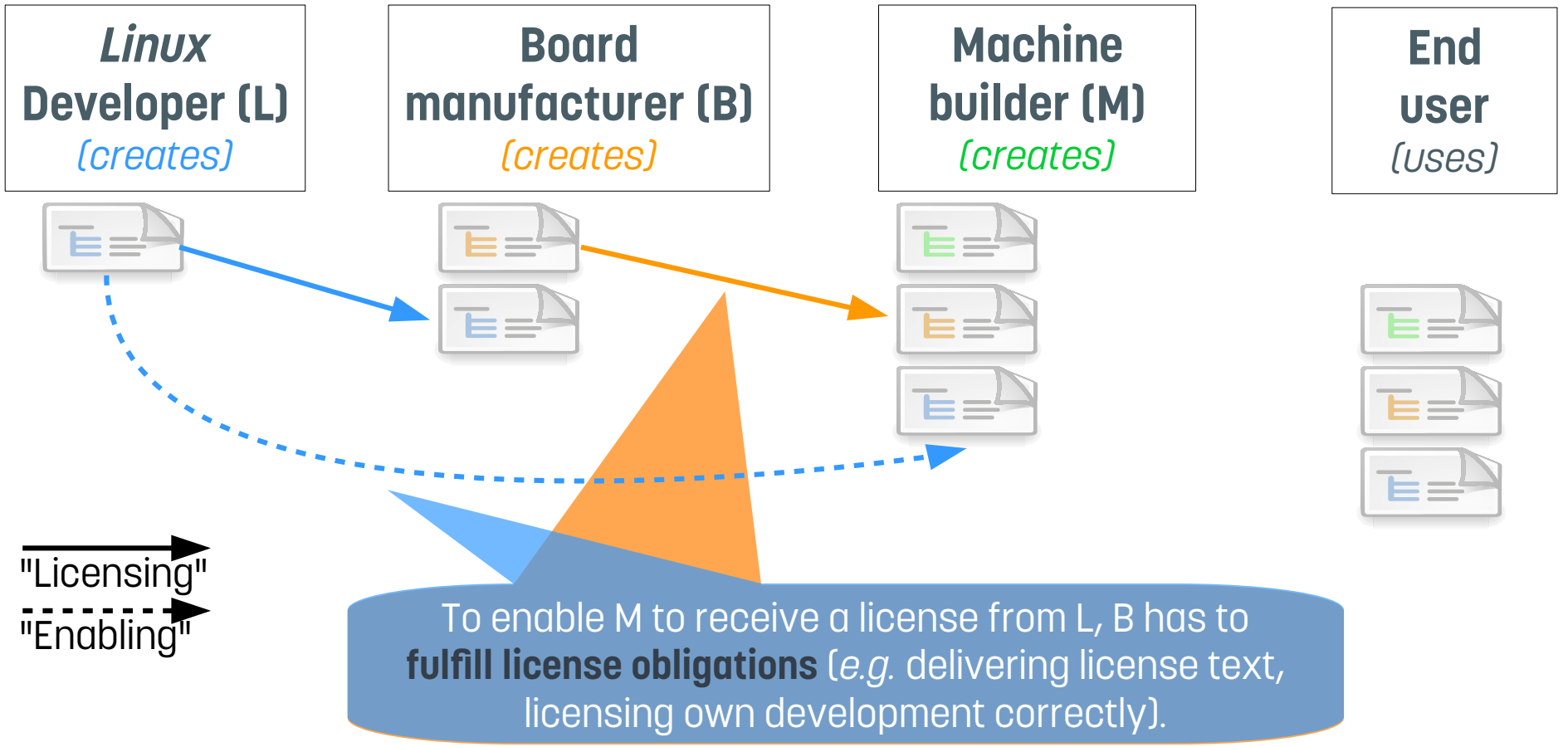
GPL software trade chain
(original licensing)



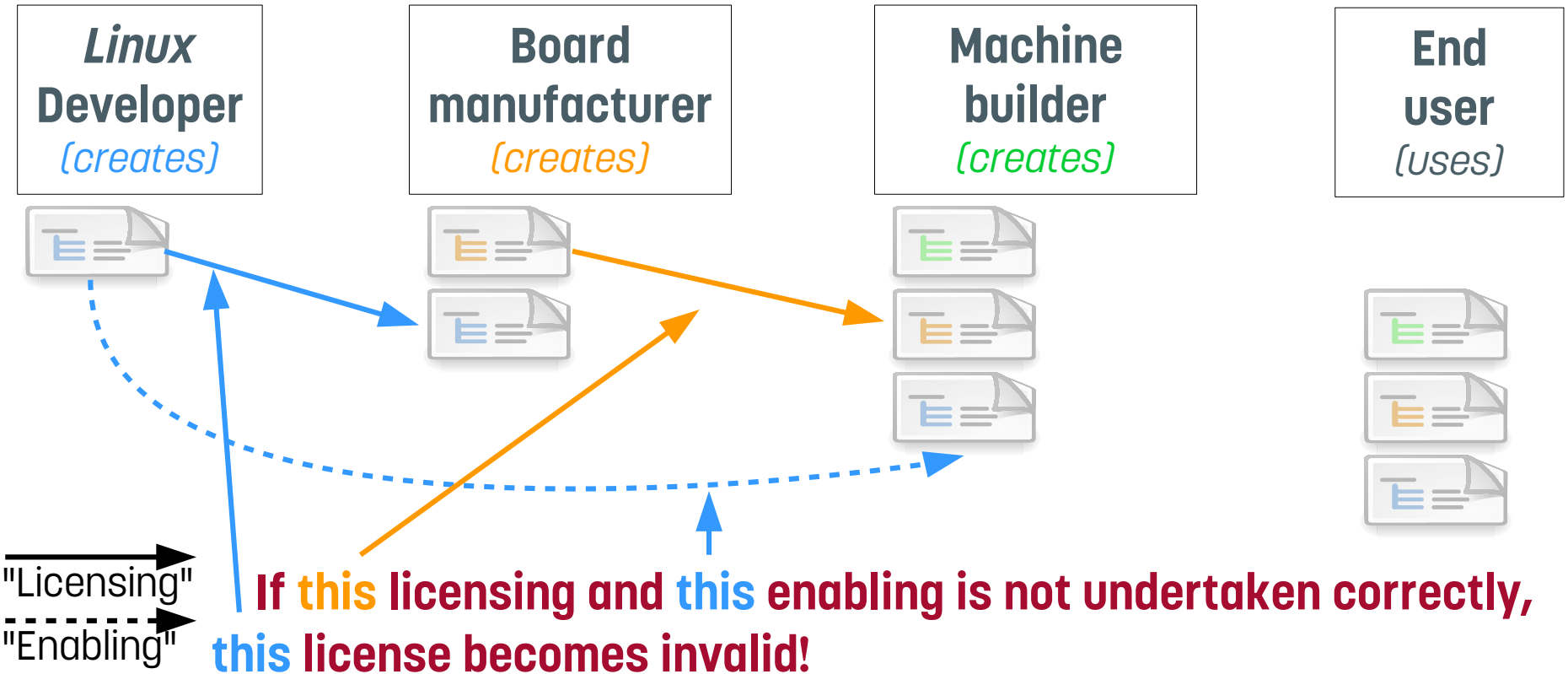
GPL software trade chain
(licensing of own development)



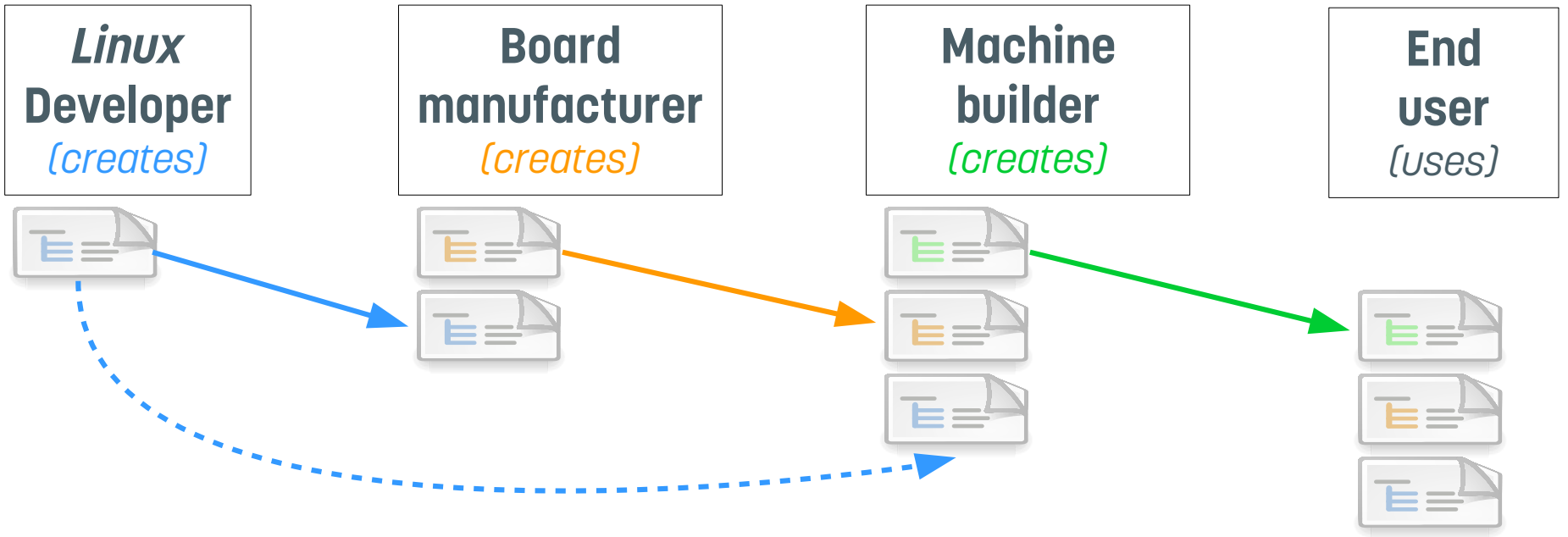
GPL software trade chain
(enabling licensing)



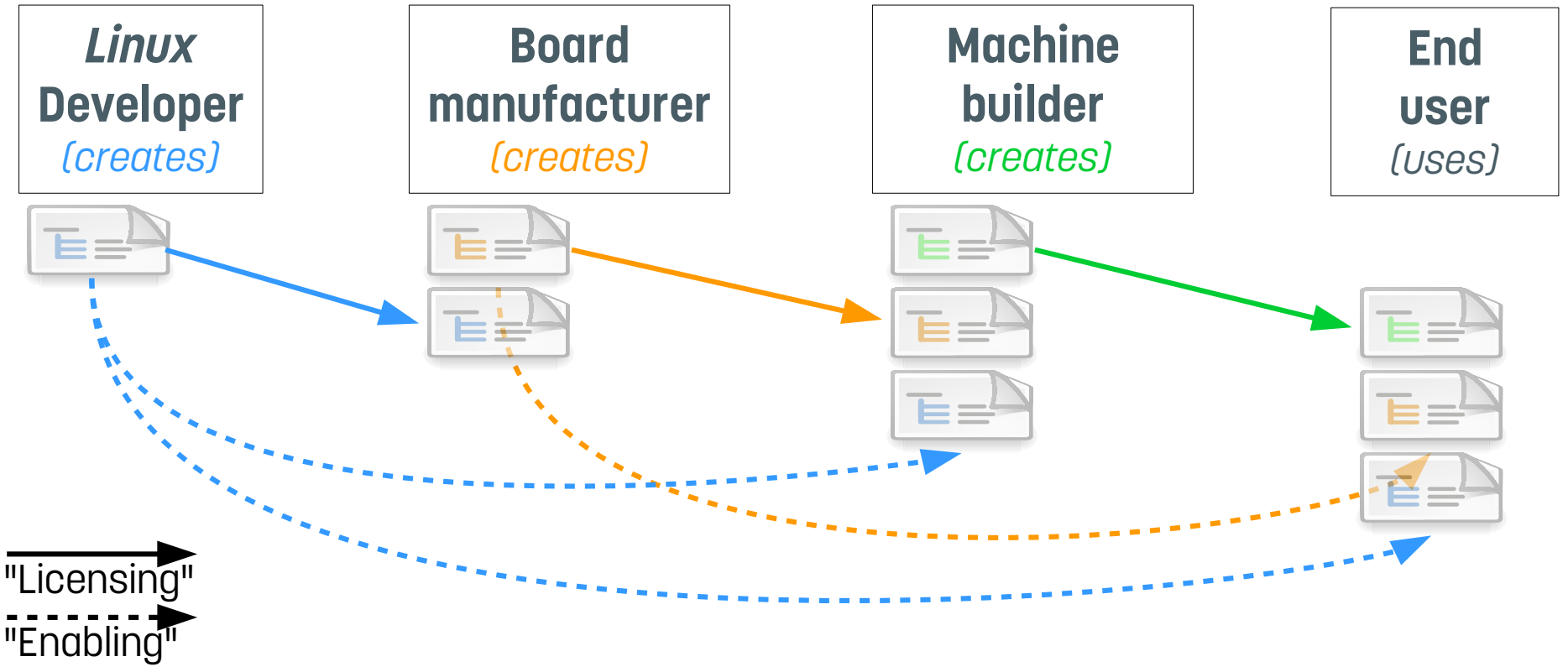
GPL software trade chain
(dependencies)



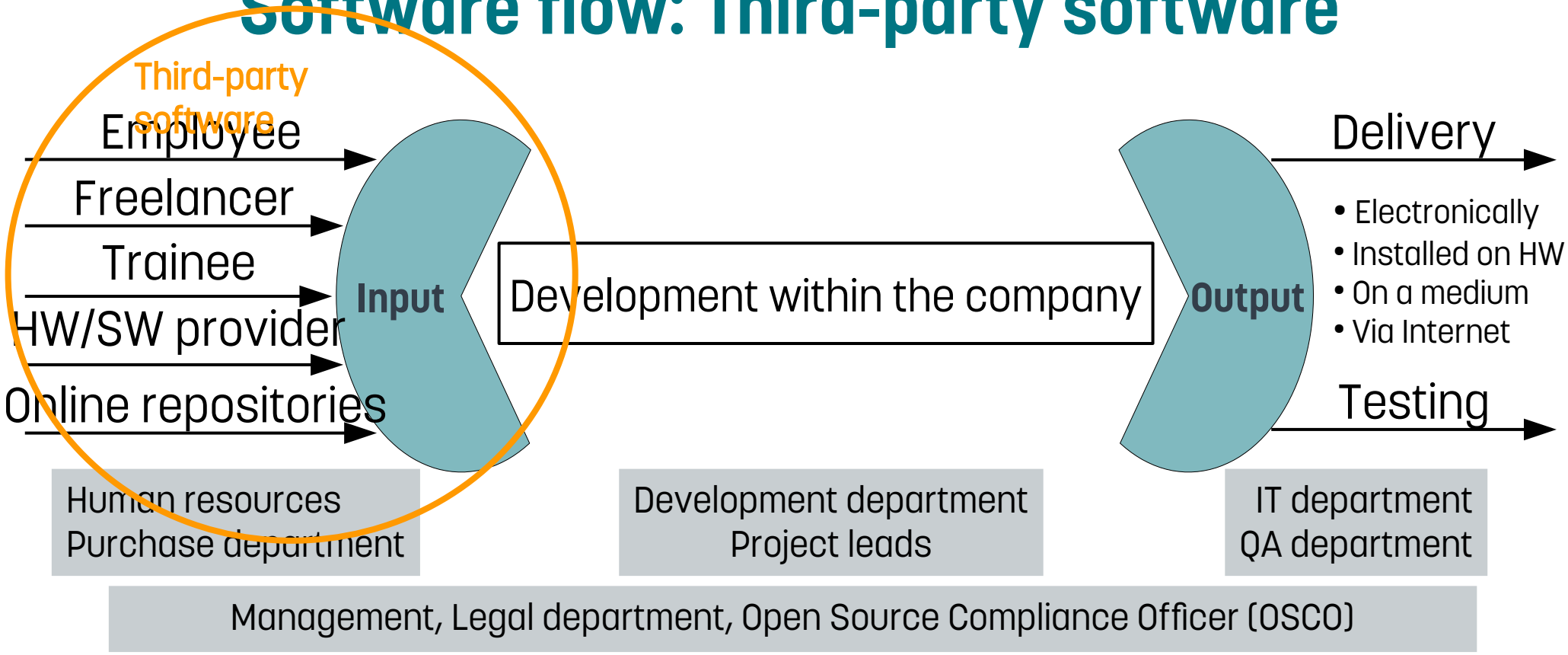
GPL software trade chain
(further steps)



GPL software trade chain
(all)



Software flow: Third-party software

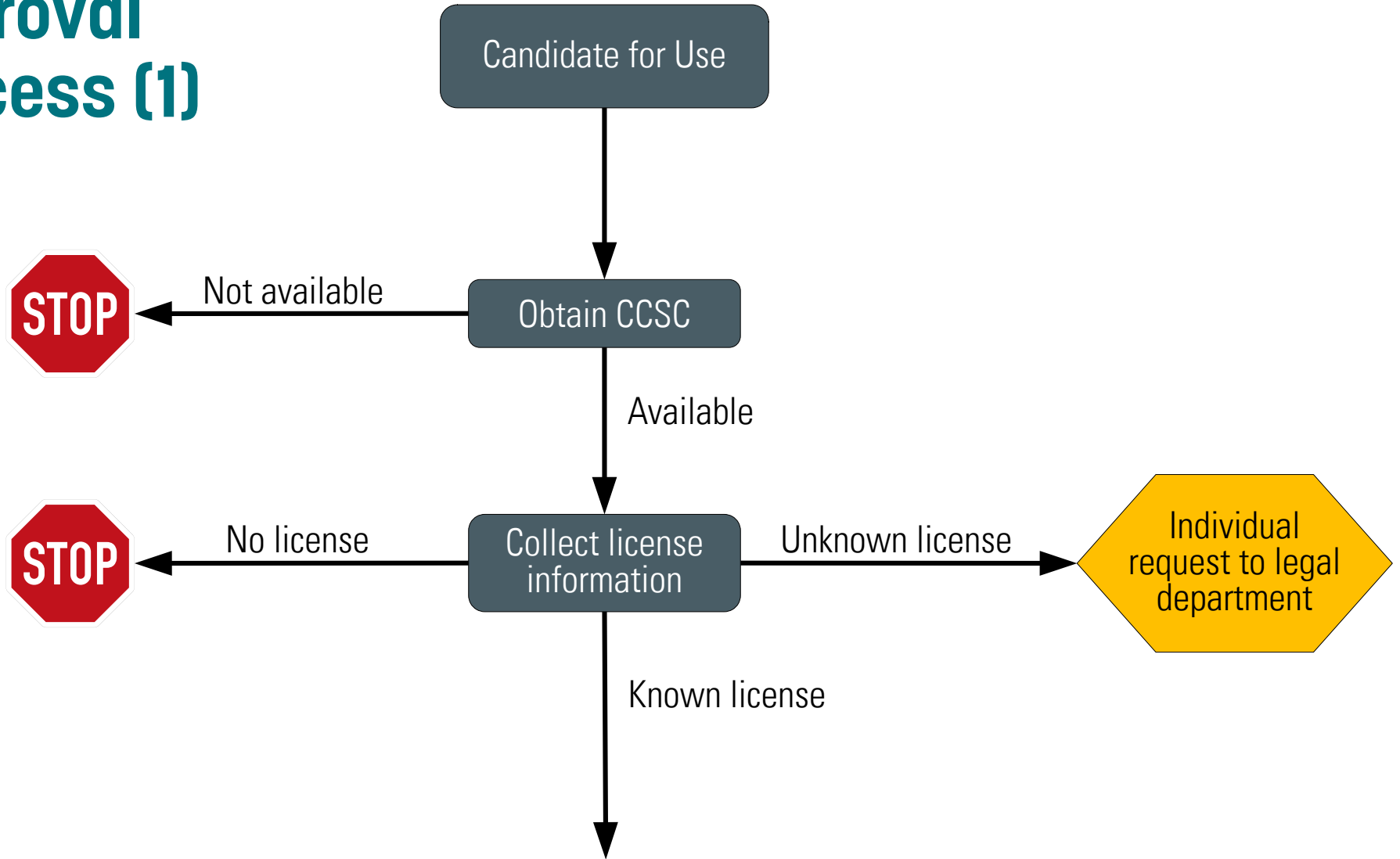


Detection and analysis of third-party software

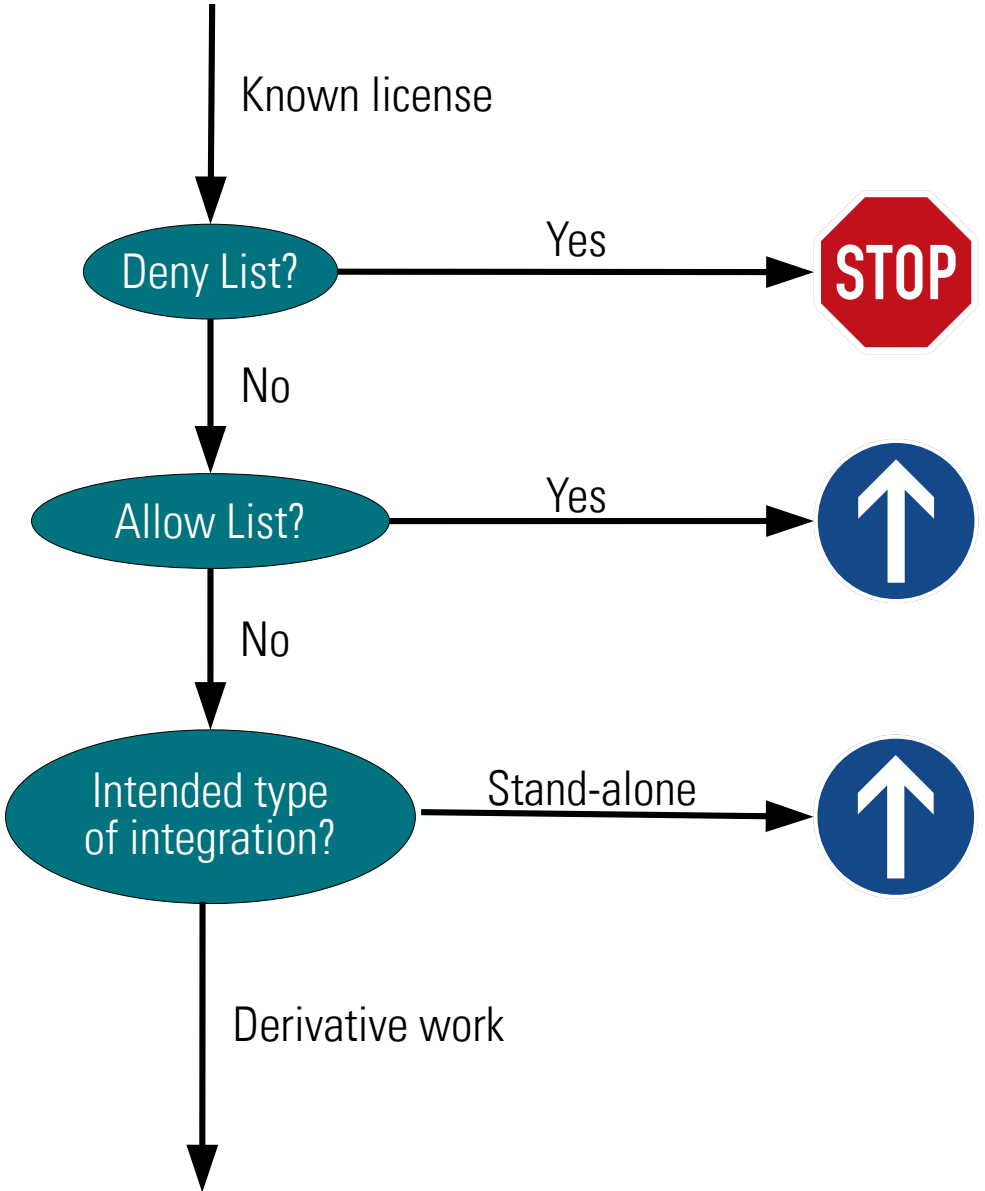
Goals

- Control over what external software is used
- Avoiding unlicensed software
- Basis for creating a BOM → Annex: Bill of Material

Approval process (1)



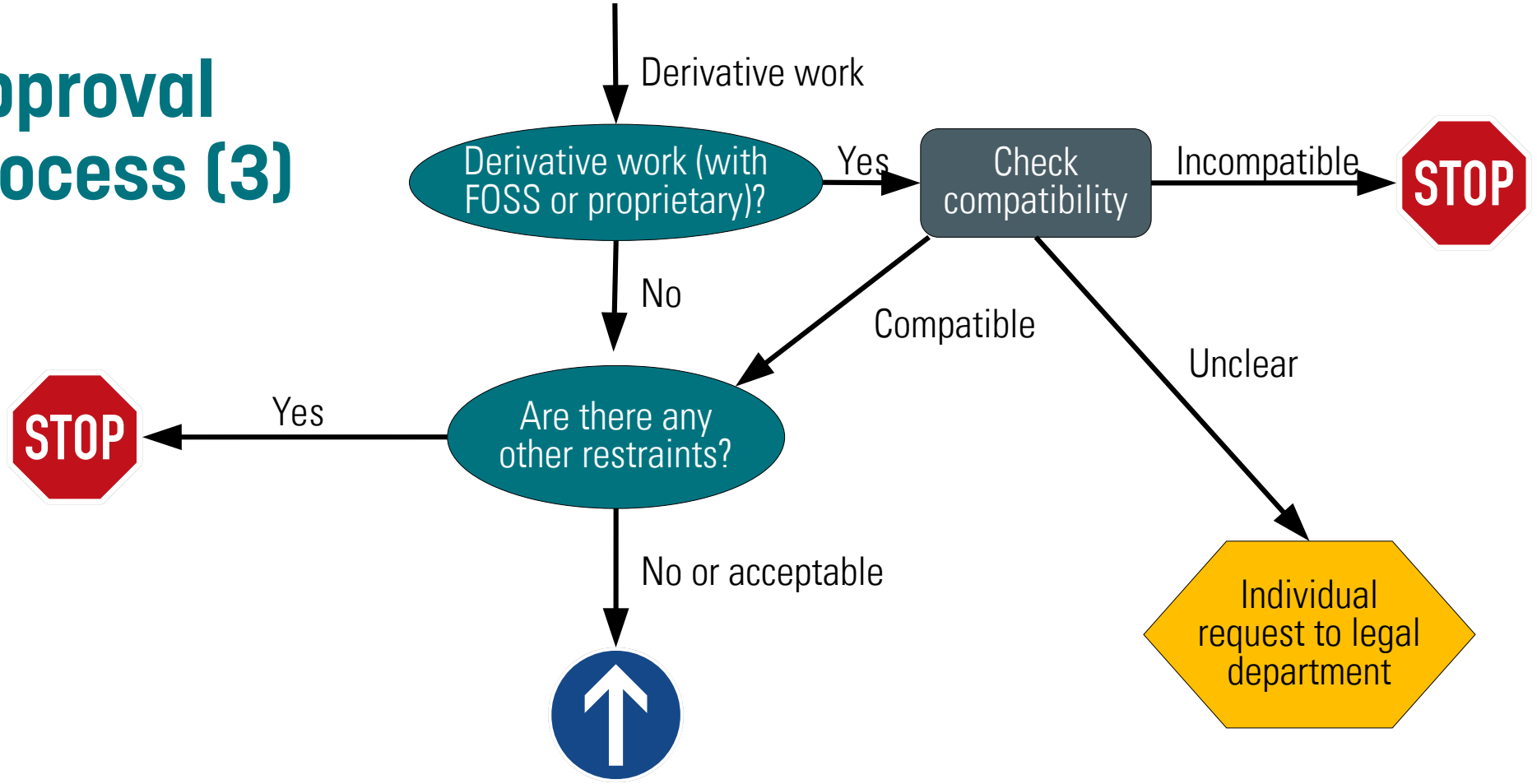
Approval process (2)



Comment: Global blocking and acceptance lists

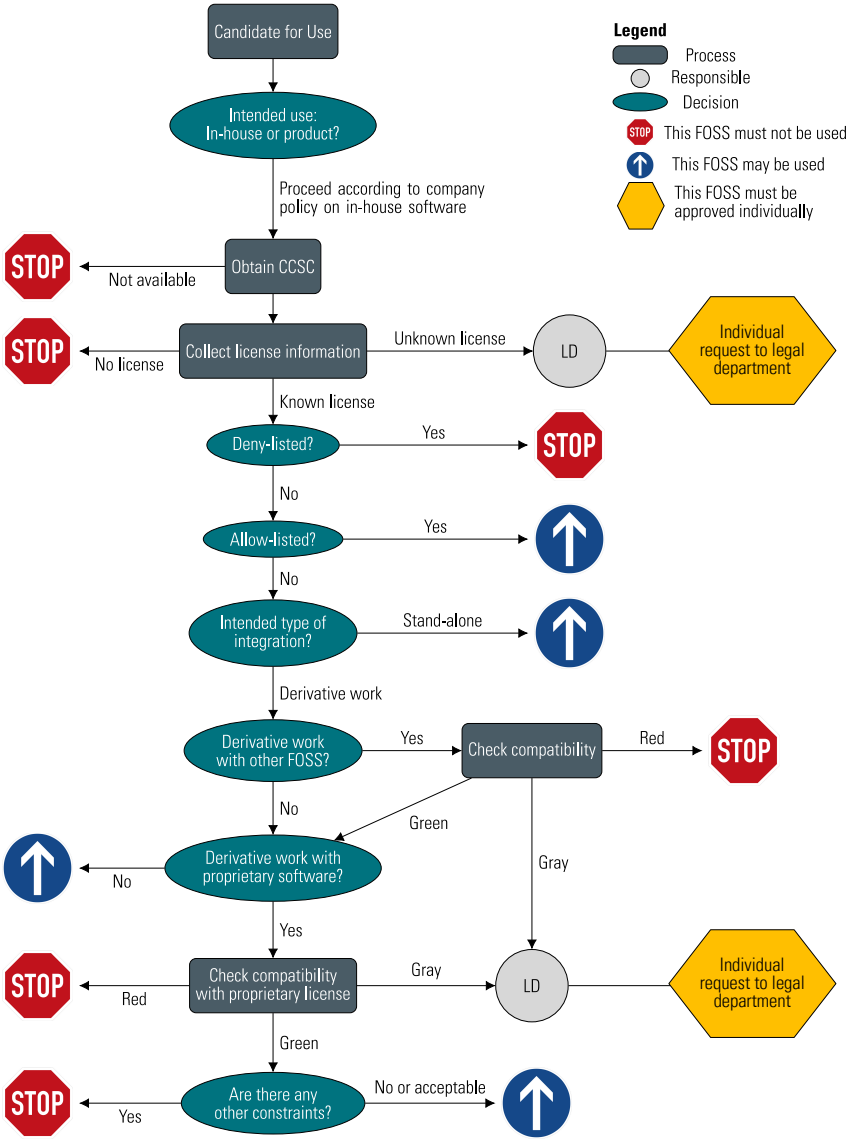
- Global blocking and acceptance lists are rarely useful.
- Licenses and their obligations must always be considered in the context of use.
- Candidates for an acceptance list are, for example, CC0-1.0 (public domain equivalent) and WTFPL, since these do not impose any obligations.

Approval process (3)



Approval process (4)

Reduces individual requests, if a software component is legally suitable for use (distribution) in a product



Murphy's Open Source Laws

1. The most inconspicuous source code causes the greatest difficulties.

Murphy's Open Source Laws

1. The most inconspicuous source code causes the greatest difficulties.
2. There is always one last source code for which you cannot obtain a license.

Murphy's Open Source Laws

1. The most inconspicuous source code causes the greatest difficulties.
2. There is always one last source code for which you cannot obtain a license.
3. The provider who promised you the moon when it came to providing licensing information is then, when you need it, incompetent or insolvent.